Section 3 Configuration

Overview . . . . . **page 3-1**
Safety Messages . . . . . **page 3-1**
Device Capabilities . . . . . **page 3-2**
General Block Information . . . . . **page 3-3**
Resource Block . . . . . **page 3-4**
Analog Input (AI) Function Block . . . . . **page 3-8**
Multiple Analog Input (MAI) Function Block . . . . . **page 3-16**
LCD Transducer Block . . . . . **page 3-18**
Advanced Diagnostics Transducer Block (ADB) . . . . . **page 3-21**
Mass Flow . . . . . **page 3-27**
Engineering Assistant Software . . . . . **page 3-28**

OVERVIEW

This section covers basic operation, software functionality, and basic configuration procedures for the Rosemount 3051S pressure transmitter with FOUNDATION fieldbus. This section is organized by block information. For detailed information about the function blocks used in the Rosemount3051S pressure transmitter, refer to “Foundation Fieldbus Block Information” on page A-1 and the FOUNDATION fieldbus Block manual (00809-0100-4783).

SAFETY MESSAGES

Procedures and instructions in this section may require special precautions to ensure the safety of the personnel performing the operations. Information that raises potential safety issues is indicated by a warning symbol (⚠). Refer to the following safety messages before performing an operation preceded by this symbol.

Warnings

⚠WARNING

Explosions can result in death or serious injury.

- Do not remove the transmitter covers in explosive environments when the circuit is live.
- Transmitter covers must be fully engaged to meet explosion proof requirements.
- Before connecting a configuration tool in an explosive atmosphere, make sure the instruments in the loop are installed in accordance with intrinsically safe or nonincendive field wiring practices.

⚠WARNING

Electrical shock can result in death or serious injury.

- Avoid contact with the leads and terminals. High voltage that may be present on leads can cause electrical shock.

DEVICE CAPABILITIES

Link Active Scheduler

The Rosemount 3051S can be designated to act as the backup Link Active Scheduler (LAS) in the event that the LAS is disconnected from the segment. As the backup LAS, the 3051S will take over the management of communications until the host is restored.

The host system may provide a configuration tool specifically designed to designate a particular device as a backup LAS. Otherwise, this can be configured manually as follows:

⚠
1. Access the Management Information Bose (MIB) for the 3051S.
2. To activate the LAS capability, write 0x02 to the BOOT_OPERAT_FUNCTIONAL_CLASS object (Index 605). To deactivate, write 0x01.

Restart the processor.

Capabilities

Virtual Communication Relationship (VCRs)

There are a total of 20 VCRs. One is permanent and 19 are fully configurable by the host system. Twenty-five link objects are available.

Network Parameter	Value
Slot Time	6
Maximum Response Delay	4
Maximum Inactivity to Claim LAS Delay	47
Minimum Inter DLPDU Delay	7
Time Sync class	4 (1ms)
Maximum Scheduling Overhead	21
Per CLPDU PhL Overhead	4
Maximum Inter-channel Signal Skew	0
Required Number of Post-transmission-gab-ext Units	0
Required Number of Preamble-extension Units	1

Host timer recommendations

T1 = 96000
T2 = 1920000
T3 = 480000

Block Execution times

Analog Input = 20 ms
PID = 25 ms
Arithmetic = 20 ms
Input Selection = 20 ms
Signal Characterizer = 20 ms
Integrator = 20 ms
Analog Output = 20 ms
Output Splitter = 20 ms
Multiple Analog Input = 20 ms
Control Selector = 20 ms

GENERAL BLOCK INFORMATION

Modes

The Resource, Transducer, and all function blocks in the device have modes of operation. These modes govern the operation of the block. Every block supports both automatic (AUTO) and out of service (OOS) modes. Other modes may also be supported.

⚠ Changing Modes

To change the operating mode, set the MODE_BLK.TARGET to the desired mode. After a short delay, the parameter MODE_BLOCK.ACTUAL should reflect the mode change if the block is operating properly.

Permitted Modes

It is possible to prevent unauthorized changes to the operating mode of a block. To do this, configure MODE_BLOCK.PERMITTED to allow only the desired operating modes. It is recommended to always select OOS as one of the permitted modes.

Types of Modes

For the procedures described in this manual, it will be helpful to understand the following modes:

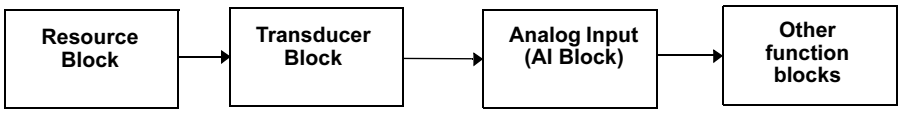
AUTO
The functions performed by the block will execute. If the block has any outputs, these will continue to update. This is typically the normal operating mode.

Out of Service (OOS)
The functions performed by the block will not execute. If the block has any outputs, these will typically not update and the status of any values passed to downstream blocks will be “BAD”. To make some changes to the configuration of the block, change the mode of the block to OOS. When the changes are complete, change the mode back to AUTO.

MAN
In this mode, variables that are passed out of the block can be manually set for testing or override purposes.

Other Types of Modes
Other types of modes are Cas, RCas, ROut, IMan and LO. Some of these may be supported by different function blocks in the Rosemount3051S. For more information, see the Function Block manual, document 00809-0100-4783.

NOTE
When an upstream block is set to OOS, this will impact the output status of all downstream blocks. The figure below depicts the hierarchy of blocks:



Block Instantiation

The Rosemount 3051S supports the use of Function Block Instantiation. When a device supports block instantiation, the number of blocks and block types can be defined to match specific application needs.The number of blocks that can be instantiated is only limited by the amount of memory within the device and the block types that are supported by the device. Instantiation does not apply to standard device blocks like the Resource, Sensor Transducer, LCD Transducer, and Advanced Diagnostics Blocks.

By reading the parameter “FREE_SPACE” in the Resource block you can determine how many blocks you can instantiate. Each block that you instantiate takes up 4.5573% of the “FREE_SPACE”.

Block instantiation is done by the host control system or configuration tool, but not all hosts are required to implement this functionality. Please refer to your specific host or configuration tool manual for more information.

Simulation

Simulation is the functionality of the AI block. To support testing, either change the mode of the block to manual and adjust the output value or enable simulation through the configuration tool and manually enter a value for the measurement value and its status (this single value will apply to all outputs). In both cases, first set the ENABLE jumper on the field device.

NOTE
All fieldbus instruments have a simulation jumper. As a safety measure, the jumper has to be reset every time there is a power interruption. This measure is to prevent devices that went through simulation in the staging process from being installed with simulation enabled.

With simulation enabled, the actual measurement value has no impact on the OUT value or the status. The OUT values will all have the same value as determined by the simulate value.

RESOURCE BLOCK

FEATURES and FEATURES_SEL

The FEATURES parameter is read only and defines which features are supported by the 3051S. Below is a list of the FEATURES the 3051S supports.

FEATURES_SEL is used to turn on any of the supported features that are found in the FEATURES parameter. The default setting of the Rosemount 3051S does not select any of these features. Choose one or more of the supported features if any.

UNICODE

All configurable string variables in the 3051S, except tag names, are octet strings. Either ASCII or Unicode may be used. If the configuration device is generating Unicode octet strings, you must set the Unicode option bit.

REPORTS

The 3051S supports alert reports. The Reports option bit must be set in the features bit string to use this feature. If it is not set, the host must poll for alerts. If this bit is set, the transmitter will actively report alerts.

SOFT W LOCK and HARD W LOCK

Inputs to the security and write lock functions include the hardware security switch, the hardware and software write lock bits of the FEATURE_SEL parameter, the WRITE_LOCK parameter, and the DEFINE_WRITE_LOCK parameter.

The WRITE_LOCK parameter prevents modification of parameters within the device except to clear the WRITE_LOCK parameter. During this time, the block will function normally updating inputs and outputs and executing algorithms. When the WRITE_LOCK condition is cleared, a WRITE_ALM alert is generated with a priority that corresponds to the WRITE_PRI parameter.

The FEATURE_SEL parameter enables the user to select a hardware or software write lock or no write lock capability. To enable the hardware security function, enable the HW_SEL bit in the FEATURE_SEL parameter. When this bit has been enabled the WRITE_LOCK parameter becomes read only and will reflect the state of the hardware switch. In order to enable the software write lock, the SW_SEL bit must be set in the FEATURE_SEL parameter. Once this bit is set, the WRITE_LOCK parameter may be set to “Locked” or “Not Locked.” Once the WRITE_LOCK parameter is set to “Locked” by either the software or the hardware lock, all user requested writes as determined by the DEFINE_WRITE_LOCK parameter shall be rejected.

The DEFINE_WRITE_LOCK parameter allows the user to configure whether the write lock functions (both software and hardware) will control writing to all blocks, or only to the resource and transducer blocks. Internally updated data such as process variables and diagnostics will not be restricted by the security switch.

The following table displays all possible configurations of the WRITE_LOCK parameter.

FEATURE_SEL HW_SEL bit	FEATURE_SEL SW_SEL bit	SECURITY SWITCH	WRITE_LOCK	WRITE_LOCK Read/Write	DEFINE_WRITE_LOCK	Write access to blocks
0 (off)	0 (off)	NA	1 (unlocked)	Read only	NA	All
0 (off)	1 (on)	NA	1 (unlocked)	Read/Write	NA	All
0 (off)	1 (on)	NA	2 (locked)	Read/Write	Physical	Function Blocks only
0 (off)	1 (on)	NA	2 (locked)	Read/Write	Everything	None
1 (on)	0 (off)(1)	0 (unlocked)	1 (unlocked)	Read only	NA	All
1 (on)	0 (off)	1 (locked)	2 (locked)	Read only	Physical	Function Blocks only
1 (on)	0 (off)	1 (locked)	2 (locked)	Read only	Everything	None

(1) The hardware and software write lock select bits are mutually exclusive and the hardware select has the highest priority. When the HW_SEL bit if set to 1 (on), the SW_SEL bit is automatically set to 0 (off) and is read only.

MAX_NOTIFY

The MAX_NOTIFY parameter value is the maximum number of alert reports that the resource can have sent without getting a confirmation, corresponding to the amount of buffer space available for alert messages. The number can be set lower, to control alert flooding, by adjusting the LIM_NOTIFY parameter value. If LIM_NOTIFY is set to zero, then no alerts are reported.

PlantWeb™ Alarms

The alarms and recommended actions should be used in conjunction with Section 2: Troubleshooting.

The Resource Block will act as a coordinator for PlantWeb alarms. There will be three alarm parameters (FAILED_ALARM, MAINT_ALARM, and ADVISE_ALARM) which will contain information regarding some of the device errors which are detected by the transmitter software. There will be a RECOMMENDED_ACTION parameter which will be used to display the recommended action text for the highest priority alarm. FAILED_ALARM will have the highest priority followed by MAINT_ALARM and ADVISE_ALARM will be the lowest priority.

NOTE
PlantWeb alerts for ADB and mass flow cannot be simulated.

FAILED_ALARMS

A failure alarm indicates a failure within a device that will make the device or some part of the device non-operational. This implies that the device is in need of repair and must be fixed immediately. There are five parameters associated with FAILED_ALARMS specifically, they are described below.

FAILED_ENABLED
This parameter contains a list of failures in the device which makes the device non-operational that will cause an alarm to be sent. Below is a list of the failures with the highest priority first.

1. Memory Failure
2. NV Memory Failure
3. Primary Value Failure
4. Secondary Value Failure
5. Sensor Module Memory Failure
6. Sensor Module Failure

FAILED_MASK
This parameter will mask any of the failed conditions listed in FAILED_ENABLED. A bit on means that the condition is masked out from alarming and will not be reported.

FAILED_PRI
Designates the alarming priority of the FAILED_ALM, see “Alarm Priority” on page 3-15. The default is 0 and the recommended value are between 8 and 15.

FAILED_ACTIVE
This parameter displays which of the alarms is active. Only the alarm with the highest priority will be displayed. This priority is not the same as the FAILED_PRI parameter described above. This priority is hard coded within the device and is not user configurable.

FAILED_ALM
Alarm indicating a failure within a device which makes the device non-operational.

MAINT_ALARMS

A maintenance alarm indicates the device or some part of the device needs maintenance soon. If the condition is ignored, the device will eventually fail. There are five parameters associated with MAINT_ALARMS, they are described below.

MAINT_ENABLED
The MAINT_ENABLED parameter contains a list of conditions indicating the device or some part of the device needs maintenance soon. If the condition is ignored, the device will eventually fail.

Below is a list of the conditions with the highest priority first.

1. Sensor Module Memory Warning
2. Primary Value Degraded
3. Secondary Value Degraded
4. Plugged Impulse Line Detected

MAINT_MASK
The MAINT_MASK parameter will mask any of the failed conditions listed in MAINT_ENABLED. A bit on means that the condition is masked out from alarming and will not be reported.

MAINT_PRI
MAINT_PRI designates the alarming priority of the MAINT_ALM, “Process Alarms” on page 3-15. The default is 0 and the recommended values is 3 to 7.

MAINT_ACTIVE
The MAINT_ACTIVE parameter displays which of the alarms is active. Only the condition with the highest priority will be displayed. This priority is not the same as the MAINT_PRI parameter described above. This priority is hard coded within the device and is not user configurable.

MAINT_ALM
An alarm indicating the device needs maintenance soon. If the condition is ignored, the device will eventually fail.

Advisory Alarms

An advisory alarm indicates informative conditions that do not have a direct impact on the device's primary functions There are five parameters associated with ADVISE_ALARMS, they are described below.

ADVISE_ENABLED
The ADVISE_ENABLED parameter contains a list of informative conditions that do not have a direct impact on the device's primary functions. Below is a list of the advisories with the highest priority first.

1. Process Anomaly Detected (SPM)
2. LOI Failure
3. PWA Simulate Active
4. Defer NV Memory Write Detected
5. Mass Flow Transducer Block Reverse Flow
6. Mass Flow Transducer Block Sensor Out of Range
7. Mass Flow Transducer Block Out of Range
8. Process Anomaly Detected (SPM)

ADVISE_MASK
The ADVISE_MASK parameter will mask any of the failed conditions listed in ADVISE_ENABLED. A bit on means the condition is masked out from alarming and will not be reported.

ADVISE_PRI
ADVISE_PRI designates the alarming priority of the ADVISE_ALM, see “Process Alarms” on page 3-15. The default is 0 and the recommended values are 1 or 2.

ADVISE_ACTIVE
The ADVISE_ACTIVE parameter displays which of the advisories is active. Only the advisory with the highest priority will be displayed. This priority is not the same as the ADVISE_PRI parameter described above. This priority is hard coded within the device and is not user configurable.

ADVISE_ALM
ADVISE_ALM is an alarm indicating advisory alarms. These conditions do not have a direct impact on the process or device integrity.

Recommended Actions for PlantWeb Alarms

RECOMMENDED_ACTION
The RECOMMENDED_ACTION parameter displays a text string that will give a recommended course of action to take based on which type and which specific event of the PlantWeb alarms is active (See Table 5-12 on page 5-10).

ANALOG INPUT (AI) FUNCTION BLOCK

Configure the AI block

⚠ A minimum of four parameters are required to configure the AI Block. The parameters are described below with example configurations shown at the end of this section.

CHANNEL

Select the channel that corresponds to the desired sensor measurement. The 3051S measures both pressure (channel 1) and sensor temperature (channel 2).

Table 3-1. I/O Channel Definitions

Channel Number	Channel Description
1	differential pressure in AI.XD_SCALE units
2	sensor temperature in AI.XD_SCALE units
3	differential pressure in inH_20 @68°F (ADB)
4	mean differential pressure
5	absolute pressure (AO.OUT)
6	process temperature (AO.OUT)
7	mass flow
8	absolute pressure (AO.CAS_IN shadow)
9	process temperature (AO.CAS_IN shadow)
10	differential pressure in inH_20 @68°F (MFTB)
11	All MAI channels (12-19 below)
12	SPM1 mean (ADB)
13	SPM1 standard deviation (ADB)
14	SPM2 mean (ADB)
15	SPM2 standard deviation (ADB)
16	SPM3 mean (ADB)
17	SPM3 standard deviation (ADB)
18	SPM4 mean (ADB)
19	SPM4 standard deviation (ADB)

NOTE
Channel 3 and channels 12-19 are only available when the Advanced Diagnostic Block is licensed. Channels 5-9 are only available when the Mass Flow Block is licensed.

L_TYPE

The L_TYPE parameter defines the relationship of the sensor measurement (pressure or sensor temperature) to the desired output of the AI Block (e.g. pressure, level, flow, etc.). The relationship can be direct, indirect, or indirect square root.

Direct
Select direct when the desired output will be the same as the sensor measurement (pressure or sensor temperature).

Indirect
Select indirect when the desired output is a calculated measurement based on the sensor measurement (e.g. a pressure measurement is made to determine level in a tank). The relationship between the sensor measurement and the calculated measurement will be linear.

Indirect Square Root
Select indirect square root when the desired output is an inferred measurement based on the sensor measurement and the relationship between the sensor measurement and the inferred measurement is square root (e.g. flow).

XD_SCALE and OUT_SCALE

The XD_SCALE and OUT_SCALE each include three parameters: 0%, 100%, and, engineering units. Set these based on the L_TYPE:

L_TYPE is Direct
When the desired output is the measured variable, set the XD_SCALE to the “Primary_Value_Range”. This is found in the Sensor Transducer Block. Set OUT_SCALE to match XD_SCALE.

L_TYPE is Indirect
When an inferred measurement is made based on the sensor measurement, set the XD_SCALE to represent the operating range that the sensor will see in the process. Determine the inferred measurement values that correspond to the XD_SCALE 0 and 100% points and set these for the OUT_SCALE.

L_TYPE is Indirect Square Root
When an inferred measurement is made based on the sensor measurement AND the relationship between the inferred measurement and sensor measurement is square root, set the XD_SCALE to represent the operating range that the sensor will see in the process. Determine the inferred measurement values that correspond to the XD_SCALE 0 and 100% points and set these for the OUT_SCALE.:

Pressure (Channel 1)	Temperature (Channel 2)	Flow (Channel 7)
Pa	°C	lbm/sec
kPa	°F	lbm/min
bar	°K	lbm/hour
mPa		lbm/day
mbar		kg/sec
torr		kg/min
atm		kg/hour
psi		grams/sec
g/cm2		grams/min
kg/cm2		grams/hour
inH_2O at 68 °F		StdCuFt/sec
mmH_2O at 68 °F		StdCuFt/min
in.H_2O at 4 °F		StdCuFt/min
mmH_2O at 4 °F		StdCuFt/hour
ftH_2O at 68 °F		StdCuFt/day
inHg at 0 °C		StdCuM/hour
mmHg at 0 °C		StdCuM/day
		NmlCuM/hour
		NmlCuM/day

NOTE
When the engineering units of the XD_SCALE are selected, this causes the engineering units of the PRIMARY_VALUE_RANGE in the Transducer Block to change to the same units. THIS IS THE ONLY WAY TO CHANGE THE ENGINEERING UNITS IN THE SENSOR TRANSDUCER BLOCK, PRIMARY_VALUE_RANGE parameter.

Configuration Examples

Pressure transmitter

Situation #1

A pressure transmitter with a range of 0 – 100 psi.

Solution

Table 3-2 lists the appropriate configuration settings.

Table 3-2. Analog Input function block configuration for a typical pressure transmitter.

.

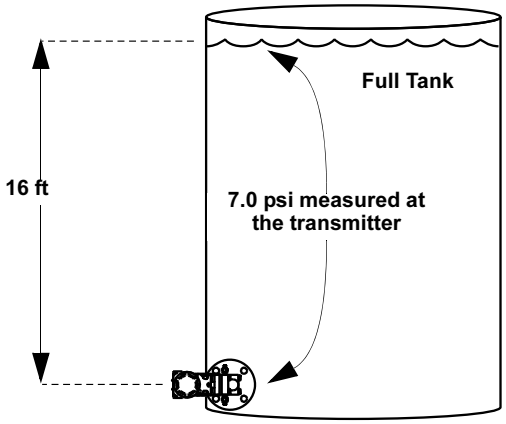
Parameter	Configured Values
L_TYPE	Direct
XD_SCALE	Primary_Value_Range
OUT_SCALE	Primary_Value_Range
Channel	1 - pressure

Pressure transmitter used to measure level in an open tank

Situation #2

The level of an open tank is to be measured using a pressure tap at the bottom of the tank. The maximum level at the tank is 16 ft. The liquid in the tank has a density that makes the maximum level correspond to a pressure of 7.0 psi at the pressure tap (see Figure 3-1).

Figure 3-1. Situation #2 Diagram.



Solution to Situation #2

Table 3-3 lists the appropriate configuration settings.

Table 3-3. Analog Input function block configuration for a pressure transmitter used in level measurement (situation #1).

Parameter	Configured Values
L_TYPE	Indirect
XD_SCALE	0 to 7 psi
OUT_SCALE	0 to 16 ft
Channel	1 - pressure

Output calculation for Situation #2
When the L_Type is configured as Indirect, the OUT parameter is calculated as:

$$OUT = \frac{PV - XD_SCALE_0\%}{XD_SCALE_100\% - XD_SCALE_0\%} * (OUT_SCALE_100\% - OUT_SCALE_0\%) + OUT_SCALE_0\%$$

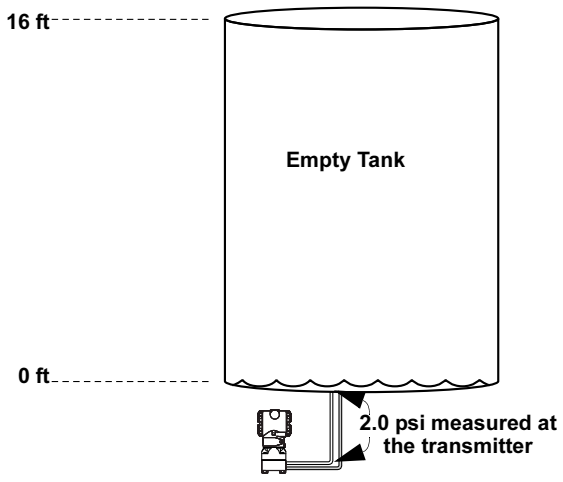
In this example, when PV is 5 psi, then the OUT parameter will be calculated as follows:

$$OUT = \frac{5\ psi - 0\ psi}{7\ psi - 0\ psi} * (16\ ft. - 0\ ft.) + 0\ ft. = 11.43\ ft.$$

Situation #3

The transmitter in situation #3 is installed below the tank in a position where the liquid column in the impulse line, with an empty tank, is equivalent to 2.0 psi (see Figure 3-2).

Figure 3-2. Situation #3 Diagram.



Solution to situation #3

Table 3-4 lists the appropriate configuration settings.

Table 3-4. Analog Input function block configuration for a pressure transmitter used in level measurement (Situation #3).

Parameter	Configured Values
L_TYPE	Indirect
XD_SCALE	2 to 9 psi
OUT_SCALE	0 to 16 ft
Channel	1 - pressure

In this example, when the PV is 4 psi, OUT will be calculated as follows:

$$OUT = \frac{4\ psi - 2\ psi}{9\ psi - 2\ psi} * (16\ ft. - 0\ ft.) + 0\ ft. = 4.57\ ft.$$

Differential pressure transmitter to measure flow

Situation #4

The liquid flow in a line is to be measured using the differential pressure across an orifice plate in the line. Based on the orifice specification sheet, the differential pressure transmitter was calibrated for 0 to 20 inH_20 for a flow of 0 to 800 gal/min.

Solution

Table 3-5 lists the appropriate configuration settings.

Table 3-5. Analog Input Function Block Configuration for a Differential Pressure Transmitter.

Parameter	Configured Values
L_TYPE	Indirect Square Root
XD_SCALE	0 to 20 in.H_2O
OUT_SCALE	0 to 800 gal/min.
Channel	1 - pressure

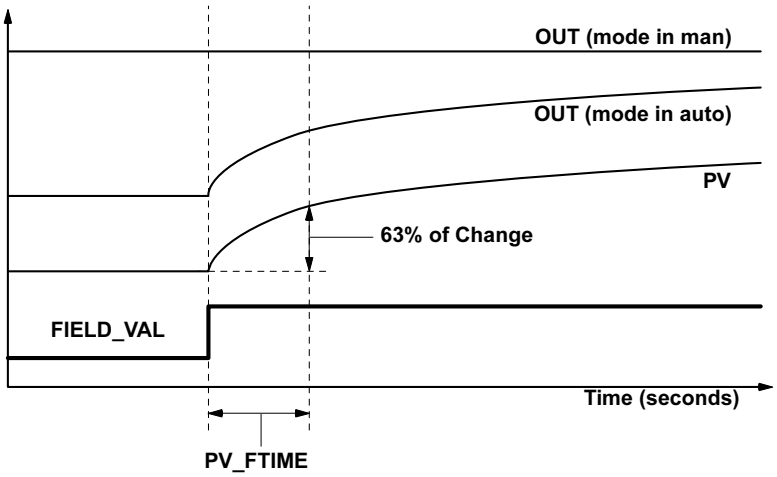
$$Out = \sqrt{\frac{PV - XDSCALE0}{XDSCALE100}}(OUTSCALE100 - OUTSCALE0) + OUTSCALE0$$

$$OUT = \sqrt{\frac{8inH_2O - 0inH_2O}{20inH_2O - 0inH_2O}}\ (800gal/min. - 0gal/min.) + 0gal/min. = 505.96gal/min.$$

Filtering

⚠ The filtering feature changes the response time of the device to smooth variations in output readings caused by rapid changes in input. Adjust the filter time constant (in seconds) using the PV_FTIME parameter. Set the filter time constant to zero to disable the filter feature.

Figure 3-3. Analog Input PV_FTIME filtering Diagram.



Low Cutoff

⚠ When the converted input value is below the limit specified by the LOW_CUT parameter, and the Low Cutoff I/O option (IO_OPTS) is enabled (True), a value of zero is used for the converted value (PV). This option is useful to eliminate false readings when the differential pressure measurement is close to zero, and it may also be useful with zero-based measurement devices such as flowmeters.

NOTE
Low Cutoff is the only I/O option supported by the AI block. Set the I/O option in **Manual** or **Out of Service** mode only.

Process Alarms

Process Alarm detection is based on the OUT value. Configure the alarm limits of the following standard alarms:

- High (HI_LIM)
- High high (HI_HI_LIM)
- Low (LO_LIM)
- Low low (LO_LO_LIM)

In order to avoid alarm chattering when the variable is oscillating around the alarm limit, an alarm hysteresis in percent of the PV span can be set using the ALARM_HYS parameter. The priority of each alarm is set in the following parameters:

- HI_PRI
- HI_HI_PRI
- LO_PRI
- LO_LO_PRI

Alarm Priority

Alarms are grouped into five levels of priority:

Priority Number	Priority Description
0	The alarm condition is not used.
1	An alarm condition with a priority of 1 is recognized by the system, but is not reported to the operator.
2	An alarm condition with a priority of 2 is reported to the operator.
3-7	Alarm conditions of priority 3 to 7 are advisory alarms of increasing priority.
8-15	Alarm conditions of priority 8 to 15 are critical alarms of increasing priority.

Status Options

Status Options (STATUS_OPTS) supported by the AI block are shown below:

Propagate Fault Forward

If the status from the sensor is Bad, Device failure or Bad, Sensor failure, propagate it to OUT without generating an alarm. The use of these sub-status in OUT is determined by this option. Through this option, the user may determine whether alarming (sending of an alert) will be done by the block or propagated downstream for alarming.

Uncertain if Limited

Set the output status of the Analog Input block to uncertain if the measured or calculated value is limited.

BAD if Limited

Set the output status to Bad if the sensor is violating a high or low limit.

Uncertain if Man Mode

Set the output status of the Analog Input block to uncertain if the actual mode of the block is Man.

NOTES
The instrument must be in **Out of Service** mode to set the status option.

Advanced Features

The AI Function Block provides added capability through the addition of the following parameters:

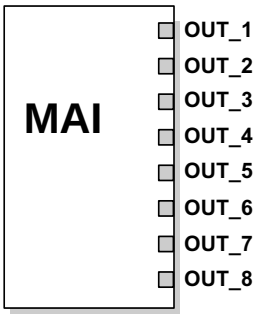
ALARM_TYPE

ALARM_TYPE allows one or more of the process alarm conditions detected by the AI function block to be used in setting its OUT_D parameter.

OUT_D

OUT_D is the discrete output of the AI function block based on the detection of process alarm condition(s). This parameter may be linked to other function blocks that require a discrete input based on the detected alarm condition.

MULTIPLE ANALOG INPUT (MAI) FUNCTION BLOCK

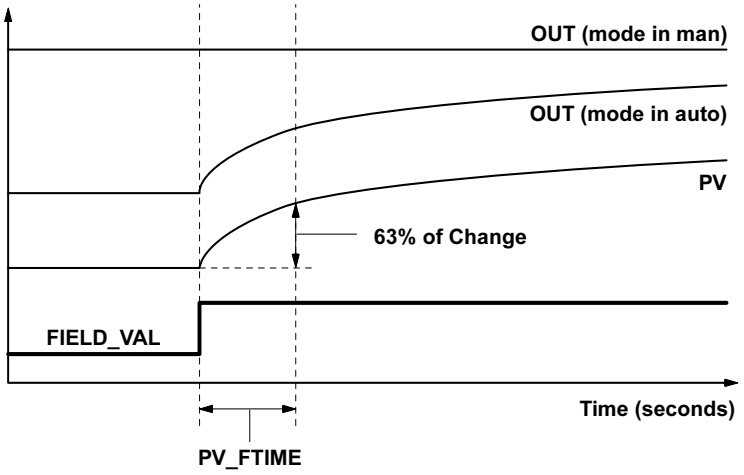


Out1 = The block output value and status for the first channel.

The Multiple Analog Input (MAI) function block has the ability to process up to eight field device measurements and make them available to other function blocks. The output values from the MAI block are in engineering units and contain a status indicating the quality of the measurement. The measuring device may have several measurements or derived values available in different channels. Use the channel numbers to define the variables that the MAI block processes.

The MAI block supports signal scaling, signal filtering, signal status calculation, mode control, and simulation. In Automatic mode, the block's output parameters (OUT_1 to OUT_8) reflects the process variable (PV) values and status. In Manual mode, values may be set manually. The Manual mode is reflected on the output status. The MAI block parameters and their units of measure, descriptions, and index numbers are listed.

Figure 3-4. Multiple Analog Input Function Block Timing Diagram



Block Errors

Table 3-6 lists conditions reported in the BLOCK_ERR parameter. Conditions in **bold** are inactive for the MAI block and are given for reference.

Table 3-6. Block Error Conditions

Number	Name and Description
0	Other
1	Block Configuration Error: the selected channel carries a measurement that is incompatible with the engineering units selected in XD_SCALE, the L_TYPE parameter is not configured, or WRITE_CHECK = zero.
2	**Link Configuration Error**
3	Simulate Active: Simulation is enabled and the block is using a simulated value in its execution.
4	**Local Override**
5	**Device Fault State Set**
6	**Device Needs Maintenance Soon**
7	Input Failure/Process Variable has Bad Status: The hardware is bad, or a bad status is being simulated.
8	Output Failure: The output is bad based primarily upon a bad input.
9	**Memory Failure**
10	**Lost Static Data**
11	**Lost NV Data**
12	**Readback Check Failed**
13	**Device Needs Maintenance Now**
14	Power Up
15	Out of Service: The actual mode is out of service.

Modes

The MAI Function Block supports three modes of operation as defined by the MODE_BLK parameter:

Manual (Man)
The block output (OUT) may be set manually

Automatic (Auto)
OUT_1 to OUT_8 reflects the analog input measurement or the simulated value when simulation is enabled.

Out of Service (OOS)
The block is not processed. PV is not updated and the OUT status is set to Bad: Out of Service. The BLOCK_ERR parameter shows Out of Service. In this mode, changes can be made to all configurable parameters. The target mode of a block may be restricted to one or more of the supported modes.

LCD TRANSDUCER BLOCK

The LCD meter connects directly to the 3051S electronics FOUNDATION fieldbus output board. The meter indicates output and abbreviated diagnostic messages.

The meter features a four-line display and a 0-100% scaled bar graph. The first line of five characters displays the output description, the second line of seven digits displays the actual value, the third line of six characters displays engineering units and the fourth line displays “Error” when the transmitter is in alarm. The LCD meter can also display diagnostic messages.

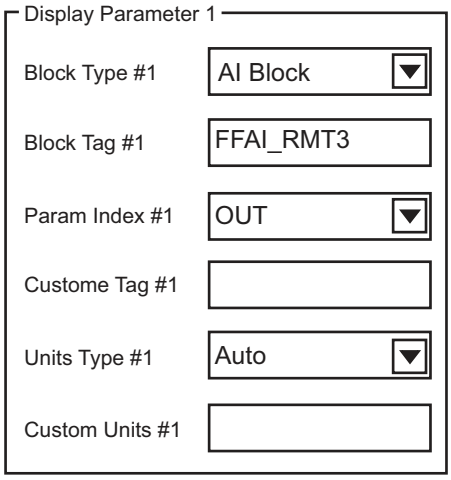
Each parameter configured for display will appear on the LCD for a brief period before the next parameter is displayed. If the status of the parameter goes bad, the LCD will also cycle diagnostics following the displayed variable:

Figure 3-5. LCD Messaging

Custom Meter Configuration

Shipped from the factory, Parameter #1 is configured to display the Primary Variable (pressure) from the LCD Transducer Block. Parameters 2 – 4 are not configured. To change the configuration of Parameter #1 or to configure additional parameters 2 – 4, use the configuration parameters below.

The LCD Transducer Block can be configured to sequence four different process variables as long as the parameters are sourced from a function block that is scheduled to execute within the 3051S pressure transmitter. If a function block is scheduled in the 3051S that links a process variable from another device on the segment, that process variable can be displayed on the LCD.



DISPLAY_PARAM_SEL

The DISPLAY_PARAM_SEL parameter specifies how many process variables will be displayed. Select up to four display parameters.

BLK_TAG_#(1)

Enter the Block Tag of the function block that contains the parameter to be displayed.

BLK_TYPE_#(1)

Enter the Block Type of the function block that contains the parameter to be displayed. This parameter is generally selected via a drop-down menu with a list of possible function block types. (e.g. Transducer, PID, AI, etc.)

PARAM_INDEX_#(1)

The PARAM_INDEX_# parameter is generally selected via a drop-down menu with a list of possible parameter names based upon what is available in the function block type selected. Choose the parameter to be displayed.

CUSTOM_TAG_#(1)

The CUSTOM_TAG_# is an optional user-specified tag identifier that can be configured to be displayed with the parameter in place of the block tag. Enter a tag of up to five characters.

UNITS_TYPE_#(1)

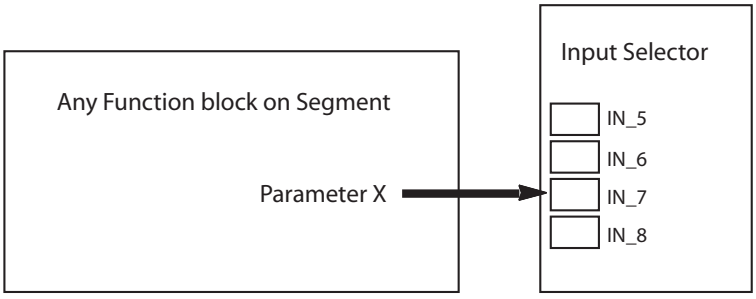
The UNITS_TYPE_# parameter is generally selected via a drop-down menu with three options: AUTO, CUSTOM, or NONE. Select AUTO only when the parameter to be displayed is pressure, temperature, or percent. For other parameters, select CUSTOM and be sure to configure the CUSTOM_UNITS_# parameter. Select NONE if the parameter is to be displayed without associated units.

(1) _# represents the specified parameter number.

CUSTOM_UNITS_#(1)

Specify custom units to be displayed with the parameter. Enter up to six characters. To display Custom Units the UNITS_TYPE_# must be set to CUSTOM.

Displaying a variable from another device on the segment (Example)



Any variable from a device on the network can be displayed on the 3051S LDC but the variable must be on a regularly scheduled communications cycle and the variable must be linked to a block within the 3051S. A typical configuration to do this is to link the output of the function block of the variable to one of the unused inputs of the Input Selector Block.

Display bar graph

The 3051S LCD is equipped with a bar graph along the top portion of the display screen. The bar graph will display the percent of range of AI.OUT (see Figure 3-6) of the AI block configured for Channel 1 (pressure) of the Sensor Transducer Block.

The bar graph on the LCD can be enabled from the DISPLAY_PARAM_SEL parameter in the LCD block.

If no AI block is found to be configured for Channel 1 the bar graph (including annunciators) will remain blank. If more than one AI block is found to be configured for the Channel 1 the AI block with the lowest OD index will be used to calculate the bar graph value.

The following equation is used to calculate the percent of range of AI.OUT:

Figure 3-6. Bar Graph Value

$$\text{Bar Graph Value} = 100 * \frac{(\text{AI.OUT} - \text{AI.OUT_SCALE @ 0\%})}{(\text{AI.OUT_SCALE @ 100\%} - \text{AI.OUT_SCALE @ 0\%})}$$

If the bar graph value calculation returns a value less than 0%, the LCD will display a bar graph value of 0%.

If the bar graph value calculation returns a value greater than 100%, then the LCD will display a bar graph value of 100%.

ADVANCED DIAGNOSTICS TRANSDUCER BLOCK (ADB)

The Advanced Diagnostics Function Block is a licensable block. It will be licensed on a new transmitter if the D01 option is ordered or can be licensed in the field via a licensing code. Please contact your local sales person on how to acquire a license for a field upgrade.

The Advanced Diagnostics Block (ADB) transducer block contains two different algorithms, Plugged Impulse Line Detection and Statistical Process Monitoring. Configuration, learning, verifying, and detection are the different phases to the Plugged Impulse Line and Statistical Process Monitoring.

ADB Parameters

For a complete listing of the Advanced Diagnostic Block (ADB) parameters see “Advanced Diagnostics Transducer Block (ADB)” on page A-13.

Plugged Impulse Line Detection (PIL)

The PIL algorithm can detect a plugged impulse line of a gage pressure (GP) or absolute pressure (AP) transmitter. For Differential pressure (DP) transmitters it can detect either single impulse line or both impulse lines plugged. This is done by comparing process dynamics the algorithm has “Learned” to a set of “Detection” parameters that are configured by the user. Some of the parameter names may appear differently depending on your host system. Configuration learning, detection are the three different phases to the Plugged Impulse Line.

IMPORTANT
Running the Advanced Diagnostics Block could affect other block execution times. We recommend the device be configured as a basic device versus a Link Master device if this is a concern.

Configuration Phase

The configuration phase is an inactive state. The configuration phase is used to set up the parameters controlling the sensitivity, thresholds, affect on PV_STATUS, length of the learning and detecting phases.

Learning Phase

The learning phase begins after a valid configuration is present. In the learning phase, the algorithm establishes a baseline of the process mean and process dynamics. This base line data is used to compare current process data to the base line when in the “Detection Phase”.

Verify Phase

The verifying phase is similar to the learning phase. It calculates a baseline value which is then compared to the learning phase baseline value. This is done to ensure a steady process before the detection phase. The amount of acceptable deviation between the two values is set up in PLINE_Learn_Sensitivity.

Detection Phase

The detection phase begins after the learning phase is complete. To determine if the impulse lines are plugged, the algorithm compares the most recent mean and standard deviation of the process to the baseline mean and standard deviation.

Configuration of the Plugged Impulse Line Detection

To configure the PIL, open the ADB transducer block and refer to the host documentation. The three parameters below are the minimum required for configuration.

PLINE_Affect_PV_Status

PLINE_Affect_PV_Status determines whether the quality of the pressure measurement from the Sensor Transducer Block will be affected by the Plugged Impulse Line status. The parameter default is “false”. When the algorithm detects plugged impulse lines there is no affect on the quality of the pressure measurement from the Sensor Transducer Block. If the parameter setting is “true”, the ADB status will cause the quality of the pressure measurement from the Sensor Transducer Block to go “UNCERTAIN”.

PLINE_Sensitivity

PLINE_Sensitivity determines the sensitivity of the PIL Detection Algorithm. Low sensitivity is primarily used if the process contains large amount of dynamics. Medium sensitivity would be recommended for normal processes. High sensitivity would be used for a process with minimal dynamics. Below is a guide for setting your “Plugging Sensitivity”.

Sensitivity levels for *all lines plugged* used for DP, AP or GP:

- **Low sensitivity**
  An 80% reduction in dynamics (with respect to the baseline dynamics) in the last detection cycle time (default 1 min.).
- **Medium sensitivity**
  A 70% reduction in dynamics (with respect to the baseline dynamics) in the last detection cycle time (default 1 min.).
- **High sensitivity**
  A 60% reduction in dynamics (with respect to the baseline dynamics) in the last detection cycle time (default 1 min.).

Sensitivity levels for *one line plugged* used for DP only:

- **Low sensitivity**
  A 100% increase in dynamics (with respect to the baseline dynamics) in the last detection cycle time (default 1 min.).
- **Medium sensitivity**
  A 70% increase in dynamics (with respect to the baseline dynamics) in the last detection cycle time (default 1 min.).
- **High sensitivity**
  A 40% increase in dynamics (with respect to the baseline dynamics) in the last detection cycle time (default 1 min).

PLINE_ON

This parameter Enables or Disables the Plugged Impulse Line Algorithm. The PIL must be both licensed and activated in order to detect plugged impulse lines. The default state is disabled.

NOTE
For most applications the above parameters are all that are needed to configure the PIL for detection.

In order to use the Advanced Set of parameters below, fully understand the underlying technology of the PIL. These parameters will fine tune the PIL to fit process conditions.

Plugged Impulse Line Advanced Settings

PLINE_Relearn

PLINE_Relearn allows the user to activate the learning phase of the PIL Detection Algorithm. To activate, “Relearn” must be selected. This can be used to reinstate the algorithm after a plugged line has been detected. The proper measures must be taken to clear the plug, and the process should be running normally before relearning the process. Relearning the process may also be necessary after changing parameters in the Advanced Set of parameters.

PLINE_Auto Relearn

PLINE_Auto Relearn determines whether or not the algorithm automatically changes to learning mode when the thresholds are exceeded. When the default “Enabled” is selected the algorithm will automatically relearn when the mean value changes by more than the Relearn Threshold value. If “Disabled” is selected, relearn must be started manually, see “PLINE_Relearn” on page 3-23.

PLINE_Relearn Threshold

PLINE_Relearn Threshold is a percent value of URL of the transmitter range. If the mean has changed by this value and Auto Relearn is “Enabled”, then the algorithm will automatically go into the learning state. If the Auto Relearn is “Disabled” this value is ignored. The default is 190 of the URL.

PLINE_Learning Cycle

PLINE_Learning Cycle determines the length of time the learning cycle uses to calculate the mean and standard deviation for the baseline. The default value is five minutes. If the process has a slow change in the mean over time, a longer learning cycle may provide a better baseline.

PLINE_Learn_Sensitivity

The PLINE_Learn_Sensitivity parameters provide very specific adjustments to the sensitivity during the learning phase.

- Insufficient Dynamics Check: Ignores the insufficient dynamic check if not selected. Use only when there is very low process noise. PIL may not detect a plugged line condition.
- 10%, 20%, and 30% Stdev. Change Check: Allows for 10, 20, or 30% change in Standard Deviation while in the learning state. If this value is exceeded, the algorithm will stay in the verifying state until the value is not exceeded.
- Three or six Sigma Mean Change Check: Allows for a three or six Sigma change in the mean while in the learning state. If this value is exceeded, algorithm will stay in the verifying state until the value is not exceeded.
- 2% Mean Change Check: The mean value of the baseline calculation can not vary more than 2% during the learning or verifying states. If this value is exceeded, algorithm will stay in the verifying state until the value is not exceeded.

PLINE_Detect_Length

PLINE_Detect_Length is the length of time the algorithm uses to take the process values and calculate the dynamics to compare to the learned baseline. This value should not be longer than the PLINE_LEARNING_LENGTH to avoid false detections. The default value is one minute. The detection cycle must be larger than the macrocycle of the system.

PLINE_Detect_Sensitivity

PLINE_Detect_Sensitivity will override the “Plugged Sensitivity” value. The values for this parameter are 0%-100% change in dynamics. Low values are used in low noise processes, and high values are used in noisy processes. This value is used when detecting two lines plugged condition for DP and line plugged for AP and GP transmitters.

PLINE_Single_Detect_Sensitivity

PLINE_Single_Detect_Sensitivity will override the Plugged Sensitivity value. The allowed values for this parameter are 0%-100% change in dynamics. Low values are used in low noise processes, and high values are used in noisy processes. This value is used when detecting single line plugged condition with DP transmitters only.

Statistical Process Monitoring (SPM)

Statistical Process Monitoring algorithm is intended to provide basic information regarding the behavior of process measurements such as PID control block and actual valve position. The algorithm can monitor up to four user selected variables. The variables must either reside in a scheduled function block which is contained in the device or be the sensor transducer block. This algorithm can perform higher levels of diagnostics by distribution of computational power to field devices. The two statistical parameters monitored by the Statistical Process Monitoring are mean and standard deviation. By using the mean and standard deviation, the process or control levels and dynamics can be monitored for change over time. The algorithm also provides:

- Configurable limits/alarms for High variation, low dynamics, and mean changes with respect to the learned levels
- Necessary statistical information for Regulatory Control Loop Diagnostics, Root Cause Diagnostics, and Operations Diagnostics.

NOTE
In Fieldbus devices, a wealth of information is available to the user. Both process measurement and control is feasible at the device level. The devices themselves contain both the process measurements and control signals that are necessary to not only control the process, but to determine if the process and control is healthy. By looking at the process measurement data and control output over time, one can gain additional insight into the process. Under some load conditions and process demands, changes could be interpreted as degradation of instruments, valves or major components such as pumps, compressors, heat exchangers, etc.... This degradation may also indicate that the loop control scheme needs to be re-tuned or re-evaluated. By learning a healthy process and continually comparing current information to the known healthy information, problems due to degradation and eventual failure can be avoided and remedied ahead of time. These diagnostics are to aid in the engineering and maintenance of the devices. False alarms and missed detections may occur. If a reoccurring problem in your process exists, please contact Rosemount for assistance with the diagnostics.

Configuration Phase

The configuration phase is an inactive state when the SPM algorithm can be configured. In this phase, the block tags, block type, parameter, limits for high variation, low dynamics, and mean change detection can be set by the user. The “Statistical Process Monitoring Activation” parameter must be set to “disabled” to configure any SPM parameter. SPM can monitor any linkable input or output parameter of a scheduled function block that resides in the device.

Learning Phase

In the learning phase of Statistical Process Monitoring, the algorithm establishes a baseline of the mean and dynamics of a Statistical Process Monitoring variable. The baseline data is compared to current data for calculating any changes in mean or dynamics of the Statistical Process Monitoring variables.

Verify Phase

The verifying phase is similar to the learning phase. It calculates a baseline value which is then compared to the learning phase baseline value. This is done to ensure a steady process before the detection phase. The amount of acceptable deviation between the two values is set up in PLINE_Learn_Sensitivity.

Monitoring Phase

The monitoring phase starts after the learning process is complete. The algorithm compares the current values to the baseline values of the mean and standard deviation. During this phase the algorithm computes the percent change in mean and standard deviation to determine if the defined limits are violated.

SPM Configuration

SPM_Bypass_Verification

If “Yes” the verification of the baseline is turned off. If “No” the learned baseline is compared to the next current calculated value to ensure a good baseline value. The recommended value is NO.

SPM_Monitoring_Cycle

SPM_Monitoring_Cycle is the length of time the process values are taken and used in each calculation. A longer monitoring cycle may provide a more stable mean value. The default is 15 minutes.

SPM#_Block_Tag

Enter the Block Tag of the function block that contains the parameter to be monitored. Block tag must be entered, there is no pull-down menu to select the tag. The tag must be a valid “Block Tag” that is in the device. The default block tags from the factory are:

AI 1400	ARITH 1900
AI 1500	INTEG 2000
AI 2100	AO2700
AI 2200	AO2800
PID 1600	OSPL2600
ISEL 1700	CSEL2500
ISEL 2400	Mass Flow 2900
CHAR 1800	

SPM can also monitor “out” parameters from other devices. To do this, link the “out” parameter to an input parameter of a function block that resides in the device, and set up SPM to monitor the input parameter.

SPM#_Block Type

Enter the Block Type of the function block that contains the parameter to be monitored.

SPM#_Parameter Index

Enter the Parameter Index of the parameter to be monitored.

Mass Flow

SPM#_Thresholds

The SPM#_Thresholds are used to allow alerts to be sent when the values are beyond the threshold values that have been set for each parameter.

Mean Limit
Alert Limit value in percent change of the Mean compared with the baseline mean value.

High Variation
Alert Limit value in percent change of the Stdev compared with the baseline Stdev value.

Low Dynamics
Alert Limit value in percent change of the Stdev compared with the baseline Stdev value.

SPM_Active

SPM_Active parameter that starts the Statistical Process Monitoring when “Enabled”. “Disabled” turns the diagnostic monitoring off. Must be set to “Disabled” for configuration. Only set to “Enabled” after fully configuring the SPM.

SPM#_User Command

Select “Learn” after all the parameters have been configured to begin the Learning Phase. The monitoring phase will start after the learning process is complete. Select “Quit” to stop the SPM “Detect” may be selected to return to the monitoring phase.

Baseline Values

The Baseline Values are the calculated values from the process over the Learning Cycle.

SPM#_Baseline_Mean
SPM#_Baseline_Mean is the calculated average of the process variable over the Learning Cycle.

SPM#_Baseline_Standard_Deviation
SPM#_Baseline_Standard_Deviation is the square root of the variance of the process variable over the Learning Cycle.

MASS FLOW

The Mass Flow Function Block is a licensable block. It will be licensed on a new transmitter if the H01 option is ordered or it can be licensed in the field via a licensing code. Please contract your local sales person on how to acquire a license.

The Mass Flow Function Block has two distinct parts to its configuration. The first part is downloading the formula the block uses to calculate mass flow. The formula is generated by using the Engineering Assistant software. The two ways to download the formula is to have the factory do it on a new order which requires you to order and fill out a C2 Configuration Data Sheet or you can do it yourself by using a National Instruments Fieldbus interface.

The second part is configuring the Mass Flow Block to accept pressure and temperature inputs from other transmitters. The differential pressure is passed to the Mass Flow Block through a channel. The pressure and temperature inputs can be linked through any FOUNDATION fieldbus configurator.

If you already own a National Instruments Fieldbus interface device, you can go to rosemount.com to download the 3051S revision 23 Device Description. If you don't already own a National Instruments Fieldbus interface device, contact your local sales person for ordering options.

NOTE
Could affect block execution times.

ENGINEERING ASSISTANT SOFTWARE

Installation and Setup

Installing the 3051S Engineering Assistant for FOUNDATION Fieldbus

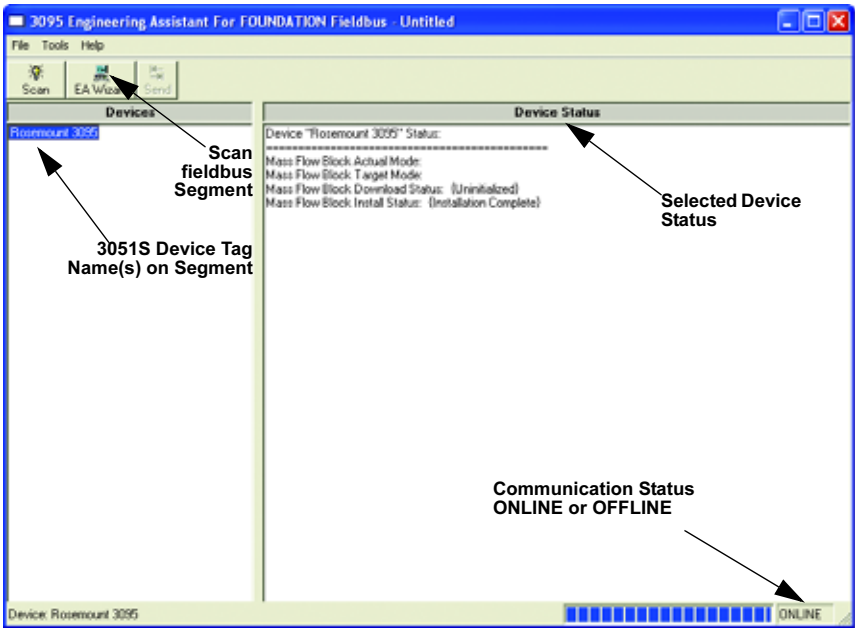
Operation of the software requires installation of both the 3051S Engineering Assistant (EA) for FOUNDATION Fieldbus program and FOUNDATION Fieldbus communication card drivers. The 3051S Engineering Assistant for FOUNDATION Fieldbus and 3051S Engineering Assistant for HART can be loaded onto the same computer. However, the programs cannot be opened simultaneously. Upgrades for the 3051S Engineering Assistant are available on www.rosemount.com.

1. Follow the instructions below to complete required software installation.
   a. Place 3051S EA Software CD-ROM disk 2 into the disk drive.
   b. Browse and select the EA-Ff folder using Windows NT, 2000, or XP.
   c. Open the ReadMe.txt file and follow the instructions provided.
2. Install the FOUNDATION Fieldbus PCMCIA communications card into the machine by following instructions provided with the card. Installation of the communications card is not required for offline operation of the 3051S EA for FOUNDATION Fieldbus software program.

Establish Communications with the 3051S FOUNDATION Fieldbus Transmitter using 3095 EA for FOUNDATION Fieldbus

1. Connect the 9-pin communications cable into the PCMCIA card port located in the computer.
2. Connect communication wiring to the cable connectors labeled "D^{+}" and "D^{-}".
3. Open the transmitter cover on the side marked "Field Terminals". Connect the communication wires to the 3051S transmitter terminals labeled "Fieldbus Wiring".
4. Verify device is properly powered to establish communications.
5. Open the 3051S Engineering Assistant for FOUNDATION Fieldbus program. Select the 3051S Engineering Assistant for FOUNDATION Fieldbus from the program menu or use the 3051S EA for FF shortcut icon.
6. Select Scan to scan the FOUNDATION Fieldbus segment. Scanning will locate and present live 3051S or 3095 FOUNDATION Fieldbus transmitters on the segment that has a licensed mass flow block. The transmitter device tag name will appear on the screen in the Device view. The Device Status view will publish the status of the transmitter.
7. FOUNDATION Fieldbus communication status is represented in the lower right corner of the screen. If status is ONLINE, communication has been established. If status is OFFLINE, communication has not been established and/or communication has been disconnected.

Figure 3-7. Device View

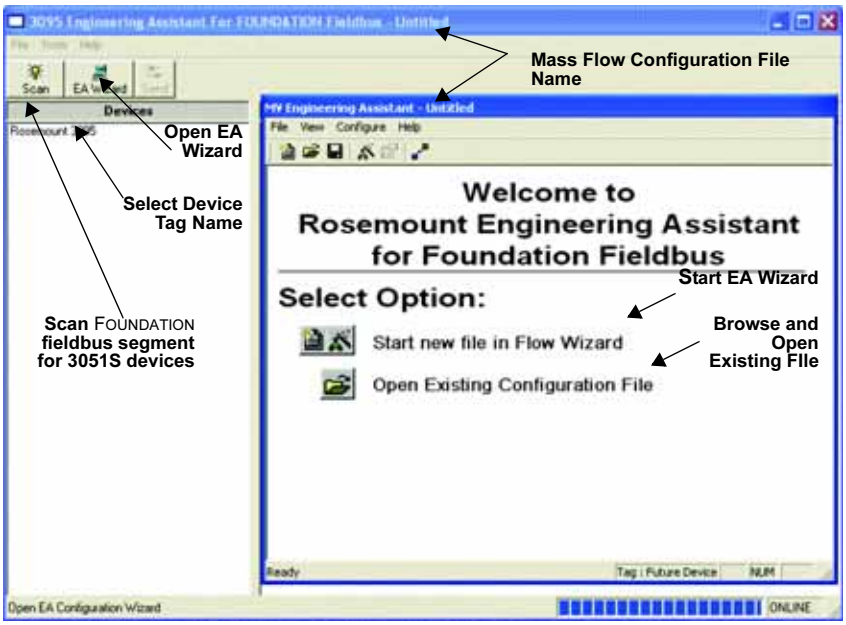


Create and Send a Mass Flow Configuration using 3051S EA for FOUNDATION Fieldbus

A mass flow configuration file can be created in either OFFLINE or ONLINE mode.

1. Select the device tag name requiring a new or updated mass flow configuration file. The selected device tag will become highlighted. Information about the selected device will appear on the Device Status portion of the screen.
2. Select the EA Wizard. A window stating, “Welcome to Rosemount Engineering Assistant for FOUNDATION Fieldbus” will appear.

Figure 3-8. Open EA Wizard



3. 3.Select either “Start new file in Flow Wizard” or “Open existing configuration files” in Flow Wizard. Select either create a new file or open a current (saved) file and edit. Follow the EA Wizard and step through completing a mass flow configuration (see page 3-33 for details.)

Figure 3-9. EA Wizard View

4. Upon completing a mass flow configuration using the EA Wizard, the file can be saved to disk. The file must be saved for review or to edit the mass flow configuration file in the future. FOUNDATION Fieldbus mass flow configuration files cannot be uploaded from the Mass Flow Transducer Block. If the file is not saved, it cannot be retrieved.
5. Select the “Send” button to download the mass flow configuration file to the Mass Flow Transducer Block. Sending the mass flow configuration file will overwrite the existing file in the Mass Flow Transducer Block. The transmitter must be out of service to send a mass flow configuration file.
6. A message box will appear confirming the action to send the mass flow configuration file to the Mass Flow Transducer Block. Select “OK” to send the mass flow configuration file.
7. Completing the download of the file to the Mass Flow Transducer Block, a screen that says “Installation Completed Successfully” will appear. Select OK.
8. The installation is now complete and will appear in the Device Status portion of the screen.
9. Bring the transmitter back into service using the host system, for example DeltaV.

Figure 3-10. Download Mass Flow Configuration File

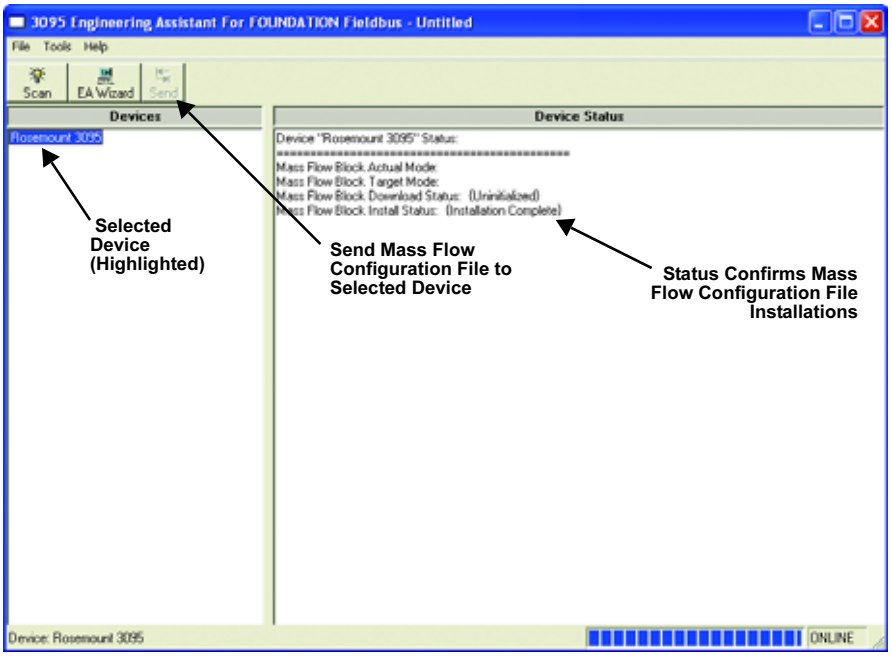


Table 3-7. Mass Flow Block Device - Specific Parameter Values

Index	Parameter	Data Type/ Structure	Store	Size	Valid Range	Initial Value	Units	Mode	Description
14	DIFFERENTIAL _PRESSURE	DS-65	D	5			inH20 @68°F	MAN	The differential pressure status and value
15	DIFFERENTIAL _PRESSURE_ SOURCE	Unsigned16	S	2	10: DP Sensor (scaled)	10	E	O/S	The channel through which the differential pressure is communicated (and converted to the proper units)
16	PRESSURE	DS-65	D	5			Psia	MAN, O/S(1)	The absolute pressure status and value
17	PRESSURE_ SOURCE	Unsigned16	S	2	5: Absolute Pressure (A0.OUT) 8: Absolute Pressure (CAS_IN shadow) 255: Constant	8	E	O/S	The channel through which the absolute pressure is communicated. When AO.OUT value is used, FAULT_STATE is used if status goes bad. When CAS_IN shadow is used, bad status/value will be propagated to the mass flow block. If a value of “Constant” is specified, the PRESSURE parameter becomes writable in O/S and the constant value is used in the calculation.
18	TEMPERATUR E	DS-65	D	5			°F	MAN, O/S(2)	The process temperature status and value
19	TEMPERATUR E_SOURCE	Unsigned16	S	2	6: Process Temperature (CAS_IN shadow) 9: Process Temperature (CAS_IN shadow) 255: Constant	9	E	O/S	The channel through which the process temperature is communicated. When AO.OUT value is used, FAULT_STATE is used if status goes bad. When CAS_IN shadow is used, bad status/value will be propagated to the mass flow block. If a value of “Constant” is specified, the TEMPERATURE parameter becomes writable in O/S and the constant value is used in the calculation.

(1) PRESSURE can be written in O/S mode if the PRESSURE_SOURCE is set to “Constant”
(2) TEMPERATURE can be written in O/S mode if the TEMPERATURE_SOURCE is set to “Constant”